RECEIVED
CENTRAL FAX CENTER
JUN 14 2004

Serial No. 10/059,088
Atty Docket AUS920010516US1

OFFICIAL

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of Michael Wayne Brown Serial No.: 10/059,088 Confirmation Number: 5267 Filed: 01/28/2002 Title: Changing the Alpha Levels of an Application Window To Indicate A Status of A Computing Task	: Before the Examiner: : Gregory F. Cunningham : Group Art Unit: 2676 : Intellectual Property Law Department : International Business Machines Corp. : 11400 Burnet Road : Austin, Texas 78758

Certificate of Facsimile Transmission

I hereby certify that this correspondence is being transmitted via facsimile to the United States Patent Office at facsimile number 703-872-9306 on June 14, 2004.

Amy Pattillo

Signature 6/14/2004 Date

APPEAL BRIEF

Mail Stop Appeal Briefs - Patents
Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

This present Brief is submitted in triplicate in support of the Appeal in the above-referenced application pursuant to a Notice of Appeal filed April 14, 2004 as required by 37 C.F.R. 1.192. This is an appeal of a final rejection dated March 11, 2004 of Claims 1-33 of application serial number 10/059,088, filed January 28, 2002.

I. Real Party in Interest

The real party in interest in the present application is the Assignee, International Business Machines Corporation of Armonk, New York, as evidenced by the Assignment set forth at Reel 012574, Frame 0078.

06/18/2004 KHARLING 00000004 090447 10059088
01 FC:1402 330.00 DA

II. Related Appeals and Interferences

Related US Patent Application Serial No. 10/059,011 is concurrently pending appeal. There are no additional Appeals or Interferences known to Appellant, Appellant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. Status of Claims

Claims 1-33 are pending. Claims 1-33 stand finally rejected as noted by the Examiner in the Examiner's Action dated March 11, 2004. Claims 7, 11, 18, 22, 29, and 33 stand objected to. Claims 1-33 are being appealed. These rejected claims which form the basis of this appeal are reproduced in the attached Appendix.

IV. Status of Amendments

A proposed amendment after final rejection was presented to the Examiner and a conference between the Examiner and Appellants' Representative Amy Pattillo was held on April 6, 2004. The Examiner stated that the proposed amendment would require a further search and therefore would not be entered. Appellants did not file the proposed amendment.

V. Summary of the Invention

As described in the present specification, the present invention discloses a method, system, and computer program for changing the alpha levels of an application window to indicate a status of a non-interactive computing task (Brown et al. page 6, lines 10-13). First, an alpha level is determined to represent a status of a non-interactive computing task (Brown et al. page 6, lines 15-17). A non-interactive task may include usage of a processor, memory, a sound card, a graphics card, a storage device, and network bandwith, for example (Brown et al. page 6, lines 17-19). Next, a transparency of at least a selected portion of a displayable object associated with the non-interactive computing task is graphically adjusted according to the alpha level (Brown et al. page 6, lines 21-23). As a result, the status of a non-interactive computing task can be displayed through an object associated with the task (Brown et al. page 6, lines 23-25).

In other embodiments, in addition to adjusting a transparency of a displayable object to represent a status of a non-interactive computing task, a color of the displayable object may also be adjusted to represent the status of the non-interactive computing task. (Brown et al. page 6, lines 28-31). In another embodiment, the transparency of a displayable object associated with an installation program may become more transparent as the program installs (Brown et al. page 7, lines 1-3). In still other embodiments, a displayable object associated with a browser may become less transparent as a page loads (Brown et al. page 7, lines 3-5).

VI. Issues Presented for Review

1. Is the Examiner's rejection of claims 1-6, 8-10, 12-17, 19-21, 23-28, and 30-32 under 35 USC 102(b) as being anticipated by Frank et al. (US Patent Number 5,651,107) well founded?

VII. Grouping of Claims

For purposes of this Appeal, claims 1-33 do not all stand or fall together.
Group I claims include claims 1, 12, and 23 which all stand or fall together.
Group II claims include claims 2, 5-6, 8-10, 13, 17, 19-21, 24, 27-28, and 30-32 which all stand or fall together.
Group III claims include claims 3, 14, and 25 which all stand or fall together.
Group IV claims include claims 4, 15, and 26 which all stand or fall together.
Group V claims include claims 7, 11, 18, 22, 29, and 33 which all stand or fall together.

VIII. Argument

Groups I, II, III, IV

Claims 1-6, 8-10, 12-17, 19-21, 23-28, and 30-32 stand rejected under 35 U.S.C. §102(b) as being disclosed by Frank et al. (US Patent Number 5,651,107) (hereinafter referred to as Frank). "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California*, 2 USPQ2d 1051, 1053 (Fed

Cir. 1987). Furthermore the reference must be an enabling disclosure of each and every element as set forth in the claim. *In re Hoecksma*, 158 USPQ 596, 600 (CCPA 1968); *In re LeGrive*, 133 USPQ 365, 372 (CCPA 1962). Because the Examiner does not show that Frank teaches each an every element of the claims 1-6, 8-10, 12-17, 19-21, 23-28, and 30-32 or enables each and every element of these claims, these claims are not anticipated and thus the rejection is not well founded, and it should be reversed.

Group I

Each of the pending claims in Group I contain the limitation of "determining an alpha level to represent a status of a non-interactive computing task". Frank does not anticipate the invention of claim 1 because Frank does not teach expressly or inherently the step of "determining an alpha level to represent a status of a non-interactive computing task." Further, Frank does not enable this step.

In the summary of the invention, Frank teaches windows having associated alpha values, where the alpha values can be selectively set and the transparency of the window adjusted according to the alpha value set for the window (Frank col. 2, lns. 56-59). However, in the summary of the invention and throughout the detailed description, the only method that Frank teaches for selectively setting the alpha value is through a user interface where the user uses a cursor control device to adjust a slider in a window to selectively set the alpha value (Frank col. 2, lns. 63-65). In addition, in Frank claim 1 teaches that "display objects have a degree of transparency determined by the transparency values associated with non-interactive display objects" which corresponds to the teaching in Frank col. 9, lns. 25-41 that a user can set an alpha value for an object by sliding a bar within the window to the alpha value. The alpha value is the transparency value and the degree of transparency of a window is determined by the alpha value (Frank col. 9, lns. 36-41).

In addition, Frank teaches windows that can have "active" regions, where the "active" region of a window can be selected to indicate that a user wants to make selections from that window without changing the position of that window in comparison with an overlaying window (Frank col. 9, ln. 63-col. 10, ln.18). Thus, in the example taught, when a first window 260 that is transparent overlays a second window, but the

user wants to make a selection from the second window without bringing the second window on top of the first window, the user selects one of the "active" regions of the second window and then can make selections within the second window without bringing the second window on top of the first window (Frank col. 9, ln. 63-col. 10, ln.18). In particular, Frank explains that "[first] window 260 not having been rendered "active" is transparent to the actions of the user, thereby permitting the user to operate on data disposed in any underlying window" (Frank col. 10, lns. 14-17).

In the Final Office Action, the Examiner cites Frank col. 9, ln. 63-col. 10, ln. 18 and claim 1 of Frank as teaching "determining an alpha level to represent a status of a non-interactive computing task." In particular, the Examiner cites the teaching of Frank of "window 260 not having been rendered "active" is transparent to the actions of the user, thereby permitting the user to operate on data disposed in an underlying window" and the teaching of Frank claim 1 where "display objects have a degree of transparency determined by the transparency values associated with each of the display objects" as teaching "an alpha level to represent a status of a non-interactive computing task" "since the user can operate on data disposed in an underlying window wherein the windows have a degree of transparency associated with each determined by the transparency values, therefore the transparency value determines via degree of transparency of the non-underlying window which not having been rendered "active" is transparent to the actions of the user "non-interactive computing task" [Final Office Action dated 3/11/2004, p. 6].

Applicants assert that claim 1 and the specification of Frank merely teaches that a display object can have a degree of transparency as determined by a transparency value associated with the display object, but Frank only teaches enabling a user to set the alpha value. In particular, Frank does not teach in claim 1 or any other part of the specification the element of determining an alpha value that represents anything, but merely teaches enabling a user to selectively adjust alpha values of windows so the graphical objects displayed in multiple windows are concurrently visible. More specifically, Frank does not teach the step of first determining an alpha value that represents the status of a non-interactive computing task and then setting the alpha value for a window associated with that non-interactive computing task to that alpha value.

Further, Applicants assert that when Frank's teaching of a "window 260 not having been rendered "active" is transparent to the actions of the user, thereby permitting the user to operate on data disposed in an underlying window" is read within the context described in Frank, Frank does not teach the detecting element of the claims of Group I. In particular, when read in context, window 260 in Figure 10 of Frank is part of a broader teaching where portions of window 260 and other windows include active window selection buttons, such that a user can select one of the active window selection buttons in a window and then operate on that window without adjusting the order of the windows overlaying one another. Thus, when read in context, Frank's teaching of operating through a non-active window is not equivalent to "determining a status of a non-interactive computing task" nor does the teaching of Figure 10 in Frank teach any method of determining a status of a "non-interactive computing task" and then computing an alpha value to represent that status.

In addition, Applicants assert that a "non-interactive computing task" is generally one that is not performed in direct response to a user input. Thus, even if Frank teaches that window 260 transparent is made transparent as a result of a user selection of an active window selection button, it is transparent as a result of an interactive computing task of selecting an active window selection button, and does not indicate the status of any non-interactive computing task associated with the application functioning within the computer in association with the transparent window. In contrast, the specification of the present invention gives an example of the difference between an interactive and non-interactive computing task. In the example, where a user selects a button associated with an audio function (e.g. selects an active window selection button taught by Frank), the interactive computing task is the actual output of the audio in response to the selection (e.g. the rendering of a window transparent in response to the selection of an active window selection button potentially taught by Frank) while the non-interactive computing tasks include at least usage of a sound card, memory and CPU, the status of which determines the transparency of the window (there is no teaching of the non-interactive computing task, the status of which determines the transparency of the window, element in Frank) (Brown et al. page 12, lines 1-5). Thus, when the teachings of Frank are compared to the teachings of the present invention, Frank does not teach

determining the status of any non-interactive computing tasks or determining an alpha value that represents the status of non-interactive computing tasks. Further, Applicants note that just because a window is non-active does not mean that non-interactive computing tasks associated with the application or other process represented by the window are not active.

In addition, Applicants assert that Frank's teaching of a user interface for user selection of an alpha value does not teach "determining an alpha level to represent a status of a non-interactive computing task." According to Frank, a user interface requires a user to select a value. Then, according to Frank, for the transparency of the window to change, the user must select another value. In contrast, the specification of the present invention discloses a computer performed step of computing an alpha level to represent a current function of the computing system and then automatically setting the transparency of a window associated with that function according to the computed alpha level. Inherent in the step is that as the status of the function changes, a new alpha value is automatically determined to represent the status and the transparency of the associated window is automatically adjusted to the new alpha value.

Furthermore, Applications respectfully propose that the allowable subject matter cited by the Examiner further shows that Frank does not anticipate the claims of Group I. The claims of Group V, are objected to by the Examiner, but would be allowed if written in independent form. Applicants respectfully propose that the claims of Group V are dependent examples of the claims in Group I and that just like the claims in Group V, the claims in Group I are not taught by Frank.

First, each claim in group I includes the element of "determining an alpha level to represent a status of a non-interactive computing task." Group V teaches applying the present invention to one type of "non-interactive computing task," a sound played in association with the displayable object, and further describes the step, "wherein said resulting transparency oscillates within said displayable object according to a frequency spectrum of a sound intended for output in associated with said displayable object." Just as Frank does not teach determining an alpha level to represent the status of the non-interactive computing task of sound, Frank does not teach determining an alpha level to represent other types of non-interactive computing tasks in general.

Next, each claim in group I includes the element of "graphically adjusting a transparency of at least a selected portion of a displayable object associated with said non-interactive computing task according to said alpha level, such that said status of said non-interactive computing task is displayed by said associated displayable object." Just like the claims in Group I, the claims in Group V teach "graphically adjusting a transparency of at least a selected portion of a displayable object". Then, the claims in Group V teach applying the present invention to another type of "non-interactive computing task," the progress of an installation program. As previous discussed, just as Frank does not teach determining an alpha level to represent the status of the non-interactive computing task of a program installation, Frank does not teach determining an alpha level to represent other types of non-interactive computing tasks in general.

Each of the pending claims in Groups I-V contain the limitation of "determining an alpha level to represent a status of a non-interactive computing task". In particular, Groups II-V contain claims that are dependent upon the claims in Group I and therefore contain the limitations of the claims in Group I. Consequently, Applicants urge that Frank fails to teach at least one element, including the element of "determining an alpha level to represent a status of a non-interactive computing task," in each of the Groups I-V claims. Therefore, reversal of the Examiner's rejection is respectfully requested.

Group III

Each of the pending claims in Group III contain the limitation of "detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and a network bandwidth." Frank does not anticipate the claims of Group III because Frank does not teach expressly or inherently element of detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and a network bandwidth." Further, Frank does not enable this element.

With respect to the claims in Group III, the Examiner cites Frank col. 2, ln. 27-col. 3, ln. 4 and in particular "text, icons, and buttons corresponding to functions to be executed by the CPU", as teaching "detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and a

network bandwidth." [Final Office Action, pages 2-3] Specifically, in the summary of the invention, Frank col. 2, ln. 27 – 38 reads as follows:

> "An apparatus and method is disclosed which has application for use in computer display systems, and in particular, display systems having object oriented graphic user interfaces with overlapping windows. A central processing unit (CPU) is provided and is coupled to a display for displaying graphic and other data in multiple overlapping windows... The windows include defined areas having window features, such as text, icons, and buttons corresponding to functions to be executed by the CPU."

In rejecting the claims of Group III, the Examiner equates the Frank teaching of "text, icons or buttons corresponding to functions to be executed by the CPU" with teaching the step of "detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and network bandwith." [Final Office Action, p. 3] Applicants respectfully propose that when the "text, icons, and buttons corresponding to functions to be executed by the CPU" are read within the context described in Frank, Frank does not teach the detecting element of the claims of Group III.

Frank col. 2, lns. 36-38 teaches a window including "defined areas *having window features, such as* text, icons, and buttons corresponding to functions to be executed by the CPU." (emphasis added) Thus, when read in context, Frank teaches a traditional window with window features that allow a user to view the results of a CPU function in "text" or invoke a function through a "button". Additionally, the specification further clarifies that the windows taught in Frank are traditional windows with selectable elements and display elements in col. 6, lns. 11-20 as follows:

> "For purposes of the present invention, a "window" may be a traditional rectangular region on a display in which data is displayed, as well as smaller sub-regions, such as pop-up, pull-down, or other menus, icons, symbols, or other display elements, and objects, generally. In the case of objects such as icons, the "data" displayed in the object may comprise only the pixels defining the icon. In objects such as rectangular windows,

menus, or sub-menus, the data displayed in such objects may include alphanumeric and/or graphic data."

Nowhere does Frank teach detecting the status of a processor, memory, sound card, graphics card, storage device, or network bandwidth. Furthermore, nowhere does Frank teach detecting the status of a processor, memory, sound car, graphics card, storage device, or network bandwidth, determining an alpha value associated with that value, and adjusting the transparency of a graphically displayed object associated with the status element according to the alpha value. Therefore, reversal of the Examiner's rejection is respectfully requested.

Group IV

Each of the pending claims in Group IV contain the limitations of "determining a color level to represent said non-interactive computing task" and "graphically adjusting said color with said transparency according to said color level of said at least said selected portion of said displayable object associated with said non-interactive computing task." Frank does not anticipate the claims of Group IV because Frank does not teach expressly or inherently the elements of "determining a color level to represent said non-interactive computing task" and "graphically adjusting said color with said transparency according to said color level of said at least said selected portion of said displayable object associated with said non-interactive computing task." Further, Frank does not enable these elements.

The Examiner specifically equates the elements of the claims in Group IV with the teaching of Frank at col. 1, lns 63-65 that reads as follows: "In systems with multiple bits, typically at least eight, it is possible to vary the intensity and color of the pixels on the display." [Final Office Action, pp. 3, 7] In fact, Frank merely teaches varying the color and intensity of windows in a display area. In contrast, the claims of Group IV teach "determining a color level to represent said non-interactive computing task" which is not taught by Frank and then adjusting the color of an associated displayable object to the color level. Further, in the same way that Frank does not teach determining an alpha level to represent a "non-interactive computing task" in the claims in Group I, Frank does not teach determining a color level to represent a "non-interactive computing task" in the

claims in Group IV. Therefore, reversal of the Examiner's rejection is respectfully requested.

Group V

Next, with regard to Group V, each of the pending claims in the Group V claims is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims. [Final Office Action, p. 5] Applicants appreciate allowance of the claims in Group V if rewritten in independent form including all of the limitations of the base claim, however, continue to assert that the claims of Group V are dependent upon base claims in Group I that should be allowed.

Distinguishing the Groups

Responsive to the requirement under 37 CFR 1.192(c)(8)(iii), Applicants assert that the claims in Groups I-V are are separately patentable.

First, the claims in Groups I-V are rejected under 35 USC 102. The claims in Group I are independent claims upon which all the other claims are dependent, and are therefore separately patentable as independent claims. The claims in Groups II-V contain claims that are dependent on the claims in Group I.

Next, the dependent claims in Group III are separately patentable because they add a limitation of the types of non-interactive computing tasks for which a status can be determined to the claims in Group I. In addition, the dependent claims in Group IV are separately patentable because they add another element to the claims in Group I of determining a color to represent a status of the non-interactive computing task and adjusting both the transparency and the color of a window to represent the status of a non-interactive computing task.

Finally, the claims in Group V are separately patentable from the claims in Groups I-IV because the claims in Group V are objected to by the Examiner, and thus would be allowed if written in independent form including all the limitations of the base claim. As such, the claims in Group V have already been identified as separately patentable by the Examiner.

CONCLUSION

It is therefore respectfully requested that the Examiner's rejection of claims 1-6, 8-10, 12-17, 19-21, 23-28, and 30-32 under 35 USC 102(b) be reversed. It is respectfully submitted that the pending claims are patentable under 35 USC 102(b) and allowance of these claims is respectfully requested.

Please charge the fee of $330.00 for submission of a Brief in Support of Appeal to IBM Corporation Deposit Account No. 09-0447. No additional filing fee is believed to be necessary; however, in the event that any additional fee is required, please charge it to IBM Corporation Deposit Account No. 09-0447.

Respectfully submitted,

6/14/2004

Amy J. Pattillo
Attorney for Applicants
Reg. No 46,983
(512) 402-9820

Appendix

1. A method for changing alpha levels of a displayable object, said method comprising the steps of:

determining an alpha level to represent a status of a non-interactive computing task; and

graphically adjusting a transparency of at least a selected portion of a displayable object associated with said non-interactive computing task according to said alpha level, such that said status of said non-interactive computing task is displayed by said associated displayable object.

2. The method for changing alpha levels of a displayable object according to claim 1, said method further comprising the step of:

graphically displaying concurrently a plurality of displayable objects independent of whether any of said plurality of displayable objects is active.

3. The method for changing alpha levels of a displayable object according to claim 1, said method further comprising the step of:

detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and network bandwidth.

4. The method for changing alpha levels of a displayable object according to claim 1, said method further comprising the steps of:

determining a color level to represent said non-interactive computing task; and

graphically adjusting said color with said transparency according to said color level of said at least said selection portion of said displayable object associated with said non- interactive computing task.

5. The method for changing alpha levels of a displayable object according to claim 1, said step of determining an alpha level further comprising the step of:

determining said alpha level according to a user preference for said transparency associated with said non-interactive computing task.

6. The method for changing alpha levels of a displayable object according to claim 1, said step of determining an alpha level further comprising the step of:

determining said alpha level, wherein said resulting transparency is uniform within said displayable object.

7. The method for changing alpha levels of a displayable object according to claim 1, said step of determining an alpha level further comprising the step of:

determining said alpha level, wherein said resulting transparency oscillates within said displayable object according to a frequency spectrum of a sound intended for output in association with said displayable object.

8. The method for changing alpha levels of a displayable object according to claim 1, said method further comprising the step of:

presenting a user within an interface for selecting transparency preferences, wherein said transparency preferences are utilized for determining said alpha level.

9. The method for changing alpha levels of a displayable object according to claim 1, said step of graphically adjusting a transparency further comprising the step of:

only graphically adjusting a transparency of transparency adjustable sections of said displayable object within said selected portion of said displayable object.

10. The method for changing alpha levels of a displayable object according to claim 1, said step of graphically adjusting a transparency further comprising the step of:

graphically adjusting a transparency of said displayable object comprising at least one of an application window, an icon, a video representation, and a graphical representation.

11. The method for changing alpha levels of a displayable object according to claim 1, said method further comprising the step of:

graphically adjusting a transparency of at least said selected portion of a displayable object associated with a progress of an installation program.

12. A system for changing alpha levels of a displayable object, said system comprising:

a graphical user interface for displaying a displayable object;

means for determining an alpha level to represent a status of a non-interactive computing task; and

means for graphically adjusting a transparency of at least a selected portion of said displayable object associated with said non-interactive computing task according to said alpha level.

13. The system for changing alpha levels of a displayable object according to claim 12, said system further comprising:

means for graphically displaying concurrently a plurality of displayable objects within said graphical user interface independent of whether any of said plurality of displayable objects is active.

14. The system for changing alpha levels of a displayable object according to claim 12, said system further comprising:

means for detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and network bandwidth.

15. The system for changing alpha levels of a displayable object according to claim 12, said system further comprising:

means for determining a color level to represent said non- interactive computing task; and

means for graphically adjusting said color with said transparency according to said color level of said at least said selection portion of said displayable object associated with said non-interactive computing task.

16. The system for changing alpha levels of a displayable object according to claim 12, said means for determining an alpha level further comprising:

means for determining said alpha level according to a user preference for said transparency associated with said non- interactive computing task.

17. The system for changing alpha levels of a displayable object according to claim 12, said means for determining an alpha level further comprising:

means for determining said alpha level, wherein said resulting transparency is uniform within said displayable object.

18. The system for changing alpha levels of a displayable object according to claim 12, said means for determining an alpha level further comprising:

means for determining said alpha level, wherein said resulting transparency oscillates within said displayable object according to a frequency spectrum of a sound intended for output in association with said displayable object.

19. The system for changing alpha levels of a displayable object according to claim 12, said system further comprising:

means for presenting a user within an interface for selecting transparency preferences, wherein said transparency preferences are utilized for determining said alpha level.

20. The system for changing alpha levels of a displayable object according to claim 12, said means for graphically adjusting a transparency further comprising:

means for only graphically adjusting a transparency of transparency adjustable sections of said displayable object within said selected portion of said displayable object.

21. The system for changing alpha levels of a displayable object according to claim 12, said means for graphically adjusting a transparency further comprising:

means for graphically adjusting a transparency of said displayable object comprising at least one of an application window, an icon, a video representation, and a graphical representation.

22. The system for changing alpha levels of a displayable object according to claim 12, said system further comprising:

means for graphically adjusting said transparency of at least said selected portion of a displayable object associated with a progress of an installation program.

23. A program for changing alpha levels of a displayable object, residing on a computer usable medium having computer readable program code means, said program comprising:

means for computing an alpha level to represent a status of a non-interactive computing task; and

means for controlling a graphical adjustment to a transparency of at least a selected portion of a displayable object associated with said non-interactive computing task according to said alpha level.

24. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for concurrently controlling a graphical display of a plurality of displayable objects independent of whether any of said plurality of displayable objects is active.

25. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for detecting said status for at least one from among usage of a processor, memory, a sound card, a graphics card, a storage device, and network bandwidth.

26. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for determining a color level to represent said non- interactive computing task; and

means for controlling a graphical adjustment of said color with said transparency according to said color level of said at least said selection portion of said displayable object associated with said non-interactive computing task.

27. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for determining said alpha level according to a user preference for said transparency associated with said non- interactive computing task.

28. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for determining said alpha level, wherein said resulting transparency is uniform within said displayable object.

29. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for determining said alpha level, wherein said resulting transparency oscillates within said displayable object according to a frequency spectrum of a sound intended for output in association with said displayable object.

30. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for controlling output of a user interface for selecting transparency preferences, wherein said transparency preferences are utilized for determining said alpha level.

31. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for controlling graphical adjustment of only a transparency of transparency adjustable sections of said displayable object within said selected portion of said displayable object.

32. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for controlling graphical adjustment of a transparency of said displayable object comprising at least one of an application window, an icon, a video representation, and a graphical representation.

33. The program for changing alpha levels of a displayable object according to claim 23, said program further comprising:

means for controlling graphical adjustment of a transparency of at least said selected portion of a displayable object associated with a progress of an installation program.

AMY J. PATTILLO
Patent Attorney
P.O. Box 161327- Austin, Tx 78716
FAX # 512-402-9820

DATE: 6/14/2004

Number of Pages to Follow (including cover sheet) : 72

SEND TO: United States Patent Office

Examiner: **Greg Cunningham**

Group Art Unit: 2676

Tel No: 703-308-6109

Fax #: 703-872-9306

FROM: Amy J. Pattillo

Tel No: 512-402-9820

Docket No. AUS92001516US1 Serial No. 10/059,088 Atty: AJP

Applicant: BROWN ET AL

X	Transmittal Letter (2 copies)	X	Certificate of Facsimile
____	Amendment	____	Notice of Appeal (2 COPIES)
____	Amendment AF	X	Appeal Brief (3 copies)
____	Ext. of Time	____	Reply Brief
____	IDS Statement	____	Change of Address
____	Other		

Deposit Acct. No. 09-0447

Fees: Amendment _____ Notice of Appeal _____ Appeal Brief $330.00 Other _____

Serial No. 10/059,088
Atty Docket AUS920010516US1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of Michael Wayne Brown Serial No.: 10/059,088 Confirmation Number: 5267 Filed: 01/28/2002 Title: Changing the Alpha Levels of an Application Window To Indicate A Status of A Computing Task	: Before the Examiner: : Gregory F. Cunningham : Group Art Unit: 2676 : Intellectual Property Law Department : International Business Machines Corp. : 11400 Burnet Road : Austin, Texas 78758

Certificate of Facsimile Transmission

I hereby certify that this correspondence is being transmitted via facsimile to the United States Patent Office at facsimile number 703-872-9306 on June 14, 2004.

Amy Pattillo

Signature 6/14/2004
Date

TRANSMITTAL OF NOTICE OF APPEAL UNDER 37 CFR 1.192(a)

Mail Stop Appeal Briefs-Patents
Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Attached is Appellants' Brief in triplicate, from a decision of the Examiner dated March 11, 2004, finally rejecting claims 1-33.

Please charge the fee of $330.00 for submission of this Appeal Brief to IBM Corporation Deposit Account No. 09-0447.

The commissioner is hereby authorized to charge any additional fee which may be required or credit any overpayment to IBM Corporation Deposit Account No. 09-0447. A duplicate copy of this document is enclosed.

Respectfully submitted,

6/14/2004

Amy J. Pattillo
Attorney for Applicants
Reg. No 46,983
(512) 402-9820